<http://en.wikibooks.org/wiki/A_Bit_History_of_Internet/Chapter_5_:_Client-Server>

Choose a low level pattern as was discussed in the text, such as shared data, pipe & filter, client server, or peer to peer. For your chosen pattern, provide a description as you might find in a software patterns book. Be sure to include items such as:

* 1. Client – Server Pattern
  2. Pattern Description – The main idea behind the client-server patter is the interaction between two computers programs, one acting as a client and the other as a server. The client typically initializes the communication cycle by requesting a service or data transformation to the server. The clients are user’s laptop, mobile devices or virtual machines running a client application. Servers are the processing node in the architecture; services devices include databases, parallel processing clusters, web servers, cloud serves, data server, services servers, which handle complex applications. In general, servers have more powerful hardware than clients such as: higher-powered central multi processors, disk drives in the Terabyte scale and more memory compare to the clients. The server will complete the client requests by sharing resources with servers in other tiers, exchanging and transforming information with processing nodes and providing the reply back to the client’s application. This pattern relies on a computer network data protocol to exchange the data reply and give the response back.
  3. Types of applications or environments:

The client-server model is the architecture behind our emails, webpages, databases requests and mobile applications. All this applications provide us with a local client front-end that communicates with a server over a network which is waiting for client’s request.

|  |  |  |
| --- | --- | --- |
| *Client Application* | *Server Application* | *Description Application* |
| HMTL-Browsers – Chrome, Firefox | Webserver- e.g. Apache | This server runs a little different than typical client-server architecture. The main webserver has the ability of to deliver the information and share it via the internet but there is third party software that is needed to display and play some information that is transmitted from the server to the client. An example of this would be a flash video. This video can be sent and saved on the web server but needs a special player (flash player) for the information to be distributed correctly on a web page. |
| Mobile applications | Google Store | The server for this plays the role on transferring short term data between the mobile/handheld devices. The client device is just used as a GUI and to display the transferred information. This is also done on a short term lease. The middleware server holds all the data and has the ability to transfer it from the existing system architecture to the new handheld architecture. |
| Online-Games- WOW | Game Service Services- Steam framework | With online games, there is a distributed architecture on the server which is known as the Coliseums. This is the design and implementation protocol that is used on the server that is used to predict different checkpoints that may be accomplished by the client also known as the player. |
| Mail Applications- Gmail , Outlook | Mail servers | The servers main use in this role is to store relay, receive and deliver mail. This is done done with a few roles of a mail server. The first way is through simple mail transfer protocol which is used for sending mail from a client. The protocol for the client to receive the mail is the post office protocol. Along with POP there is one other for receiving mail at the client and this is called the Internet Message Access Protocol. |
| Database applications- Banking, etc… | Data Servers/ Cloud Servers | The server is used to store the shared data so multiple clients can connect and add information to the database. This makes the information consistent across all of the client computers in the model. |
| FTP-Client e.g Filezilla client | Ftp Server e.g Filezilla server | Filezilla Server, Interface used for the remote client to connect and perform file transfers in both directions. |

* 1. Advantages/disadvantages or strengths/weaknesses of this pattern:

Advantages:

* *Parallelism* - Server side is able to handle multiple clients in parallel
* *Flexibility* - Server can manage different types of clients (e.g., different browsers)
* *Reliability* - Infrastructure reliability with , Cache server and Backup serves

Disadvantages:

* Relies on a network infrastructure to function properly
* May be more exposed to some security issues
* Multiple request at the same time may cause lag in the whole system
  1. Client – Server Diagram

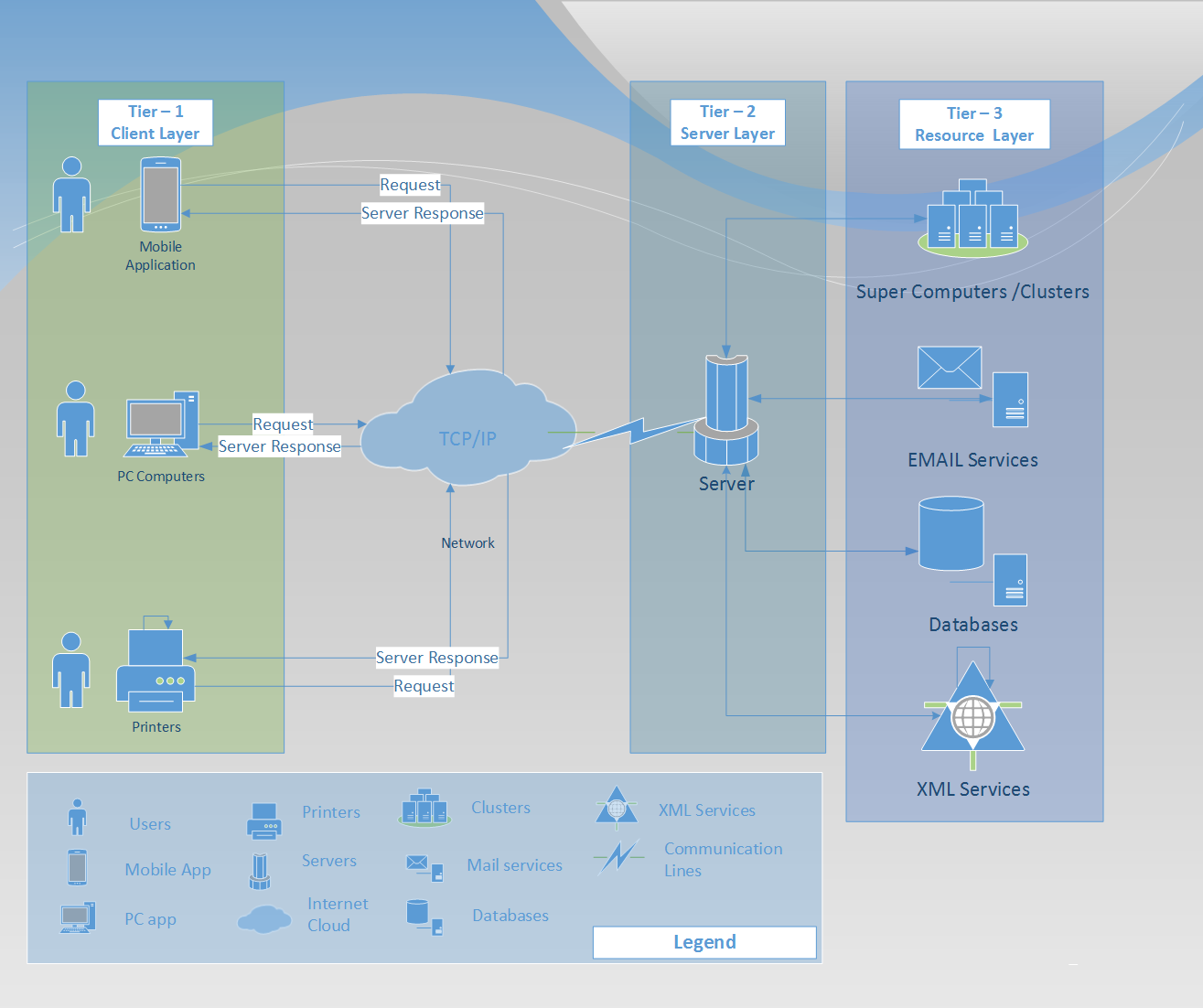


Figure 1- Client-Server diagram

* 1. Additional Comments: Tiers-N, Cache, Middleware